

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-325867

(43) 公開日 平成9年(1997)12月16日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 3/12			G 0 6 F 3/12	B
				T
B 4 1 J 5/30			B 4 1 J 5/30	Z
29/38			29/38	Z

審査請求 未請求 請求項の数11 O L (全 12 頁)

(21) 出願番号 特願平8-143193

(22) 出願日 平成8年(1996)6月5日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 島田 宗毅

東京都大田区下丸子3丁目30番2号 キヤ  
ノン株式会社内

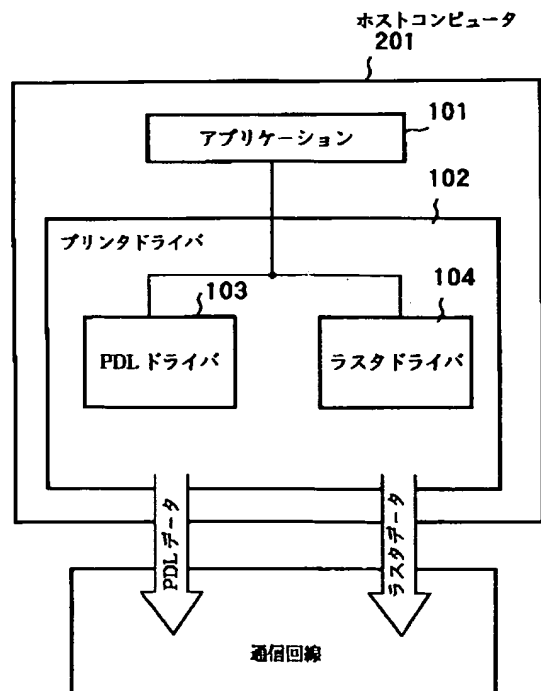
(74) 代理人 弁理士 大塚 康徳 (外1名)

(54) 【発明の名称】 印刷制御方法及び印刷システム

(57) 【要約】

【課題】印刷出力を高画質にかつ効率良く行う。

【解決手段】アプリケーション101で作成された画像データは、所定の描画命令の形式で記述されている。プリントドライバ102では、その描画命令に関して、PDLで記述できるか判定し、できなければラスタドライバ104を用いてラスタ形式にデータを変換してプリンタに送信する。また、PDLで記述可能な場合には、その描画命令をラスタデータに変換するのに要する時間と、PDLに変換されたデータをプリンタでレンダリングするのに要する時間とを予測計算する。それらの内、短い時間で可能と見積もられた形式にデータを変換し、PDLドライバ103あるいはラスタドライバ104を用いてプリンタに送る。この場合のプリンタは、ラスタデータを受け取るとそれを印刷し、PDLデータを受け取るとそれをビットマップ画像に展開して印刷する。



## 【特許請求の範囲】

【請求項1】 所定形式の描画命令で記述された画像データを、ラスタ形式及びページ記述言語形式の印刷を出力する印刷装置から印刷出力させる印刷制御方法であって、

前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測工程と、

ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測工程と、

前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換工程と、

前記変換工程により変換されたデータを前記印刷装置に送信する送信工程とを備えることを特徴とする印刷制御方法。

【請求項2】 前記変換工程によりページ記述言語形式に変換された元の描画命令を保持する保持工程を更に備え、前記第1及び第2の予測工程は、前記描画命令の各々に対して順次予測を行って第1及び第2の時間それぞれを積算し、前記変換工程は、前記第1の時間が第2の時間より短くなった時点で、前記保持工程により保持されている描画命令に遡ってラスタ形式に変換を行うことを特徴とする請求項1に記載の印刷制御方法。

【請求項3】 前記第1及び第2の予測工程は、ページを単位として、各ページに含まれる描画命令すべてについて第1及び第2の時間を予測し、前記変換工程は、前記第1及び第2の予測工程により時間の予測されたページについて、予測結果に基づいて変換を行うことを特徴とする請求項1に記載の印刷制御方法。

【請求項4】 前記描画命令がページ記述言語に変換できるか判定する判定工程を更に備え、該工程により変換できないと判定された場合には、前記変換工程により描画命令をラスタ形式に変換することを特徴とする請求項1乃至3のいずれかに記載の印刷制御方法。

【請求項5】 前記変換工程によりラスタ形式に変換されたデータを、前記送信工程により送信する前に圧縮する圧縮工程を更に備えることを特徴とする請求項1乃至4のいずれかに記載の印刷制御方法。

【請求項6】 ホスト装置において所定形式の描画命令で記述された画像データを印刷装置から印刷出力させる印刷システムであって、

前記ホスト装置は、前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測手段と、

ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測手段と、

前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換手段と、

前記変換工程により変換されたデータを前記印刷装置に送付する手段とを備え、

前記印刷装置は、前記ホスト装置から受信したラスタ形式あるいはページ記述言語形式の印刷データを受信する手段と、

受信したデータがページ記述言語の形式である場合に、それを展開する展開手段と、

ページ記述言語から展開されたデータあるいはラスタ形式のデータを印刷出力する出力手段とを備えることを特徴とする印刷システム。

【請求項7】 前記変換手段によりページ記述言語形式に変換された元の描画命令を保持する保持手段を更に備え、前記第1及び第2の予測手段は、前記描画命令の各々に対して順次予測を行って第1及び第2の時間それぞれを積算し、前記変換手段は、前記第1の時間が第2の時間より短くなった時点で、前記保持手段により保持されている描画命令に遡ってラスタ形式に変換を行うことを特徴とする請求項6に記載の印刷システム。

【請求項8】 前記第1及び第2の予測手段は、ページを単位として、各ページに含まれる描画命令すべてについて第1及び第2の時間を予測し、前記変換手段は、前記第1及び第2の予測手段により時間の予測されたページについて、予測結果に基づいて変換を行うことを特徴とする請求項6に記載の印刷システム。

【請求項9】 前記描画命令がページ記述言語に変換できるか判定する判定手段を更に備え、該手段により変換できないと判定された場合には、前記変換手段により描画命令をラスタ形式に変換することを特徴とする請求項6乃至8のいずれかに記載の印刷システム。

【請求項10】 前記変換手段によりラスタ形式に変換されたデータを、前記送信手段により送信する前に圧縮する圧縮手段を更に備えることを特徴とする請求項6乃至9のいずれかに記載の印刷システム。

【請求項11】 所定形式の描画命令で記述された画像データを、ラスタ形式及びページ記述言語形式の印刷を出力する印刷装置から印刷出力させるプログラムを格納するコンピュータ可読メモリであって、

前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測工程のコードと、

ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測工程のコードと、

前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換工程のコードと、

前記変換工程により変換されたデータを前記印刷装置に送信する送信工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は印刷制御方法及び印刷システムに関し、特にCAD（計算機支援用設計）、CG（コンピュータグラフィックス）、デザイン、ビジネスにおけるDTP（デスクトップパブリッシング）分野等のマルチメディア処理で利用されるデータを、高品位・高速に印刷・記録する印刷制御方法及び印刷システムに関する。

#### 【0002】

【従来の技術】従来、実用的に用いられる画像プリンタでは、ホストコンピュータから送られてきたページ記述言語（PDL）により記述されたデータを解析し、ラスターデータに変換してメモリにレンダリングして排紙していた。転送用データとして、ラスターデータに比べて一般にサイズの小さなPDLデータを用いることによって、低速な伝送回線におけるトラフィックのボトルネックを回避し、プリンタ内蔵された高速な描画ハードウェアによる高速なレンダリングを行い、高速なページ出力を行うことを可能としていた。また、従来（MS-DOS等）のテキストベースのOSにおいて様々なアプリケーションによって用いられていた、それぞれのアプリケーション固有の描画方法をプリント出力の際にはPDLのコマンドに合わせた形にする意味でもPDLは有効であった。

【0003】しかし、ここ数年ホストコンピュータのハード・ソフト両面での機能の革新は著しく、以下の3つの理由により、上記PDLの利点は一部失われてきている。1. グラフィカルなユーザインターフェースを有するウィンドウシステムをベースとしたOSがホストコンピュータで標準的に動作するようになり、ドローイングソフト等のアプリケーションはグラフィックスの描画を行う際、それぞれのウィンドウシステムが持つ描画のためのライブラリ関数（グラフィックスインターフェース）を利用するようになった。表示された画像をそのまま印刷するいわゆるWYSIWYG（What You See Is What You Get）な印刷を行うためには、このグラフィックス関数で描画されたものと全く同じものをPDLを用いてプリンタで描画する必要があり、グラフィックス関数が行う手続を全て同等の機能を持つPDLに変換せねばならない。一般的に、OSの持つグラフィックスインターフェースの描画機能はPDLの描画機能よりも高機能である。また、グラフィックスインターフェースの機能はホストコンピュータのグラフィックス計算処理性能の向上とともに日進月歩で革新される。

【0004】このため、グラフィックスインターフェースの機能とPDLの機能とを対応付けていくことは非常に難しく、グラフィックスインターフェースが複雑な描画処理を行う場合には、画面上と異なる不正な描画出力となるケースがある。また、グラフィックスインターフェースの持つ高度な機能をプリミティブなレベルのPDLコマンドに変換することで、発行するコマンド数が増

え、処理速度が低下するケースが多くなってきた。2. ホストコンピュータの処理能力が飛躍的に向上して、プリンタ側でラスターデータに展開するよりも、ホスト側で展開した方が高速である場合がある。3. ホストとプリンタを接続する通信が高速になってきたことや、データ圧縮技術の進歩によって、PDLとしてデータを送る時間とページラスターデータを送る時間とに差がなくなってきた。

【0005】上記の理由により、PDLデータによるプリンタへのデータ転送は、ウィンドウシステムを持つOS上で動作するアプリケーションのデータを印刷する際に、出力が不正確、あるいは非効率な場合がある。そこでプリンタドライバでは、PDLを用いることで出力が不正確または非効率になる場合は、プリンタドライバの設定を切り替え、PDLデータではなくホスト側で印刷データを最終的に出力する形にラスターライズされたデータとして、プリンタに転送する。この時のラスターライズはOSのグラフィックスインターフェースを用いて行われるので、描画不正は発生しない。

#### 【0006】

【発明が解決しようとする課題】従来、上述のPDL出力とラスターライズデータ出力の変更を行うためには、ユーザが手動でプリンタドライバの出力方法を切り替えることが必要であった。ここでは、PDLデータを出力するドライバを「PDLドライバ」、ラスターデータを出力するドライバを「ラスタードライバ」と呼ぶ。このとき、PDLドライバを選択した場合の問題として、出力が不正になる、あるいはラスタードライバでの出力と比べて出力時間の長い場合がある、等が挙げられる。また、ラスタードライバを選択した場合の問題として、PDLドライバでの出力と比べて出力時間が長い場合がある、ホストコンピュータのデスクなどの資源を多く消費する、等が挙げられる。

【0007】この切り替えの判断をユーザが行うことは難しいため、結果的に非効率な方のドライバを選択することも多かった。また、様々なタイプのグラフィックスインターフェースコマンドが組み合わされて描画が行われた場合に、どちらのドライバを選択しても相互排反的な非効率になる場合があった。

【0008】本発明は上記従来例に鑑みてなされたもので、印刷しようとするデータに適した形式で印刷データを印刷装置に送り、効率の良い印刷処理を行なえる印刷制御方法及び印刷システムを提供することを目的とする。

#### 【0009】

【課題を解決するための手段】上記目的を達成するために本発明の印刷制御方法及び印刷システムはつぎのような構成から成る。すなわち、所定形式の描画命令で記述された画像データを、ラスター形式及びページ記述言語形式の印刷を出力する印刷装置から印刷出力させる印刷制

御方法であって、前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測工程と、ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測工程と、前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換工程と、前記変換工程により変換されたデータを前記印刷装置に送信する送信工程とを備える。

【0010】また、本発明の印刷システムは次のような構成から成る。すなわち、ホスト装置において所定形式の描画命令で記述された画像データを印刷装置から印刷出力させる印刷システムであって、前記ホスト装置は、前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測手段と、ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測手段と、前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換手段と、前記変換工程により変換されたデータを前記印刷装置に送付する手段とを備え、前記印刷装置は、前記ホスト装置から受信したラスタ形式あるいはページ記述言語形式の印刷データを受信する手段と、受信したデータがページ記述言語の形式である場合に、それを展開する展開手段と、ページ記述言語から展開されたデータあるいはラスタ形式のデータを印刷出力する出力手段とを備える。

【0011】また、本発明のコンピュータ可読メモリはつぎのような構成から成る。すなわち、所定形式の描画命令で記述された画像データを、ラスタ形式及びページ記述言語形式の印刷を出力する印刷装置から印刷出力させるプログラムを格納するコンピュータ可読メモリであって、前記画像データをラスタ形式に変換するための第1の時間を予測する第1の予測工程のコードと、ページ記述言語形式に変換された前記画像データを前記印刷装置により展開するための第2の時間を予測する第2の予測工程のコードと、前記第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換工程のコードと、前記変換工程により変換されたデータを前記印刷装置に送信する送信工程のコードとを備える。

【0012】

【発明の実施の形態】以下、添付図面に沿って本発明に関わる実施例を詳細に説明する。

〔第1の実施の形態〕図1、図2は実施例の「印刷装置」の基本構成を示すブロック図である。この図を用いて処理の大まかな流れを示す。〔全体構成〕図1は、ホストコンピュータ201の内部構成を示す。アプリケーション101から出力されたグラフィックスインターフェースを利用した描画データは、プリンタドライバ10

2によって処理され、プリンタへの出力データが生成される。プリンタドライバ102はPDLドライバ103とラスタドライバ104に大きく内部的に分けられる。アプリケーション101からプリンタドライバ102に対して転送されるグラフィックスインターフェースのデータは、PDLドライバ103を用いてPDLデータに変換、あるいはラスタドライバ104を用いてラスタデータに展開することができる。ホストコンピュータ201は、作成したデータをプリンタに様々な通信回線を用いて転送する。

【0013】図2にプリンタ221の構成図を示す。ここで、ホストコンピュータ201とコントローラ214間にPDLデータまたはラスタデータが流れる。通信形態はシリアル、ネットワーク、バス接続等何であっても問題はないが、パフォーマンス的には高速通信路であることが望ましい。

【0014】送られたデータは入力バッファ202に格納される。

【0015】転送されたデータがPDLデータの場合には、プログラムROM206内のPDLコマンド解析プログラムによって入力データがスキャンされる。フォントROM203は、文字のビットパターンまたはアウトライン情報、及び文字ベースラインや文字メトリック情報を格納し、文字の印字に際して利用される。パネルIOP204は、プリンタ本体に装着されるパネルにおけるスイッチ入力の検知やLCDへの表示を司るI/Oプロセッサ及びファームウェアであり、低価格のCPUが利用される。拡張I/F205は、プリンタの拡張モジュール（フォントROM、プログラムROM、RAM、ハードディスク等）とのインターフェース回路である。

【0016】プログラムROM206はプリンタソフトウェアを格納するROMであり、ROM206に格納されたプログラムをCPU212が実行することでプリンタ221全体を制御して、転送データを読み込み処理を実行する。管理用RAM207はソフトウェアのための管理領域であり、入力されたPDLを解析して中間データ形式（オブジェクト）に変換したデータや、グローバル情報等がRAM207に格納される。

【0017】ハードレンドラ208は、レンダリング処理をASICハードウェアで実行することにより、プリンタエンジン213（LBP）のビデオ転送に同期して実時間でレンダリング処理を行い、少ないメモリ容量でのバンド処理を実現するものである。ページバッファ209は、PDL言語によって展開されるイメージを格納する領域である。

【0018】プリンタインターフェース210はプリンタエンジン213との間で、ページバッファ209の内容をプリンタ側の水平・垂直同期信号に同期して、ビデオ情報として転送する。本インターフェースではプリンタとの間に、プリンタへのコマンド送信やプリンタから

のステータス受信を行う。

【0019】転送されたデータがラスターデータである場合には、ハードレンドラ208を用いることなく、転送データを直接ページバッファに描画する。

【0020】CPU212はプリンタコントローラ内部の処理を制御する演算装置である。プリンタエンジン213は、コントローラから送出されるビデオ信号を印字し、コントローラ214は本処理を実現するプリンタを制御する。

〔ホストによる処理の流れ〕図1、図2の中での矢印は各種描画情報に関する処理の流れを示す。この処理の流れのうち、ホストコンピュータにおける処理の手順を図3のフローチャートを用いて説明する。なお、図1におけるアプリケーション101やプリンタドライバ102を実行するために、ホストコンピュータ201の構成は図5のようになっている。

【0021】すなわち、アプリケーションやドライバ等は、ハードディスクや光ディスク、フロッピーディスク等の2次記憶505に格納されており、実行時に読み出されて主メモリ502にロードされる。CPU501は主記憶上のプログラムを実行して種々のアプリケーションなどを処理する。また、操作者はディスプレイ503及びキーボード・マウス504により操作を行う。プリンタとの接続は、I/Oインターフェース506を介して行う。図3、図4の処理手順もまたCPU501により実行されるプログラムとして実現される。

【0022】図3、図4は、プリンタドライバ102により行われる処理の一部であり、PDLドライバとラスタードライバといずれを用いるか選択して印刷出力を行なわせるものである。

【0023】まず、ステップS301においてプリンタへ転送するデータの作成をホストコンピュータ上で開始する。ステップS302では、アプリケーション101より出力されるグラフィックインターフェースの描画コマンドを受け取る。

【0024】ステップS303においては、アプリケーションから出力された描画コマンドをドライバで変換してプリンタへ転送する際に、PDLデータとして転送した方が良いのか、ラスターデータとして転送した方が良いのかを判定する。ステップS303の判定アルゴリズムの詳細を図4に示す。

【0025】まず、ステップS401においてグラフィックスイインターフェースの転送する描画コマンドが、PDLでは描画不可能なものであるかを判定する。

【0026】PDLで描画不可能とは、処理対象の描画コマンドを、それをホストコンピュータでラスターライズした場合と正確に同じ結果を産み出すPDLに変換することができない、ということである。PDLで描画不可能な描画コマンドは、それらの組合わせしだいで予め知ることができる。そのため、例えば、プリンタドライバ

のインストール時、あるいはアプリケーションのインストール時など、描画コマンドとPDLとの組み合わせが変わる場合に、組合せごとにPDLで描画不可能な描画コマンドを登録しておく。そうすることで、PDLでは描画不可能として登録された描画コマンドを参照してステップS401における判定を下すことができる。

【0027】PDLで描画不可能と判定されれば、ステップS307へ進み、ラスタードライバに処理を切替える。

【0028】PDLで描画可能と判定された場合は、次にステップS402へ進み、発行されたグラフィックスインターフェースコマンドをPDLに置き換えた場合に、プリンタによってどの位の描画時間がかかるのかを予測計算する。予測計算には、各PDLコマンドを様々なパラメータで実行した場合の計算コストを示す、予め準備された情報を利用する。この情報に、処理対象となっている描画コマンドのパラメータを適用した場合の計算コストを描画に必要な時間として予測する。予測計算された結果は、いままで計算された予測描画時間の累積に加算する。

【0029】次にステップS403において、発行されたグラフィックスインターフェースコマンドをラスタードライバを利用してホストコンピュータでラスターライズする時間を予測計算する。予測計算には、様々なパラメータで送られる各グラフィックスインターフェースコマンドをラスターライズした場合の計算コストを示す、予め準備された情報を利用する。この場合にも、ステップS402と同様、処理対象となっている描画コマンドのパラメータを適用して時間を予測する。予測計算された結果は、いままで計算されたラスターライズ予測時間の累積に加算する。なお、ラスター形式のデータを圧縮してプリンタに送付し、プリンタで伸長する場合には、その圧縮・伸張に要する時間も予測計算する。

【0030】ステップS404において、PDL予測描画時間累積とグラフィックスインターフェースコマンドラスターライズ予測描画時間累積とを比較し、ラスターライズ予測時間の方が短い場合、ステップS307へ進み、ラスタードライバへ処理を切替える。

【0031】一方、PDL描画時間の方が短い場合にはステップS304へ進み、PDLデータを作成する。

【0032】ステップS305では、変換されたグラフィックスインターフェースコマンドを、メモリあるいはディスク上にスプールする。これは、この後に転送されるグラフィックスインターフェースコマンドに描画不可能なコマンドが含まれていた場合に、再び始めから全てのコマンドをラスタードライバで解析しなければならないからである。

【0033】ステップS306において、アプリケーションから転送される描画コマンドが完了したかをチェックし、完了していると判定されれば生成されたPDLデ

ータをプリンタに転送する。完了していなければ次の描画コマンドについて処理を施す。

【0034】一方、ステップS307へと進み、ラスタドライバへの切替えが発生した場合、PDLデータの生成を中止し、それまでに生成されているPDLデータを消去する。プリンタドライバ内部でPDLドライバからラスタドライバへの切替えを行い、ラスタデータとしてプリンタへデータ転送を行う処理に切り替える。

【0035】ステップS308において、スプールされた全てのグラフィックスインターフェイスコマンドをラスタドライバによって解析し、ラスタデータを生成する。また、一旦ラスタドライバに切り替えたなら、それ以降は、アプリケーションから転送される全てのデータについて、ラスタデータを生成する。

【0036】こうして変換されたPDLデータ、あるいはラスタデータを、ステップS309でプリンタに転送する。この際、ラスタデータは、イメージなど必要に応じて圧縮される。プリンタでは、圧縮されている場合にはそのデータを伸張圧縮ハードウェア211により伸長する。

【0037】以上の手順により、ホストコンピュータ上のアプリケーションで作成された画像を印刷出力する際、ラスタデータあるいはPDLデータのどちらの形式のデータとして印刷データをプリンタに送信するかを決定し、決定された形式に変換してデータをプリンタに送信する。このため、データに応じて最適な形式を選ぶことができ、印刷効率を向上させることができる。〈エンジンの構成〉ここで、プリンタに用いられるエンジンの構成の例を説明しておく。

【0038】図6はエンジン213として用いられるレーザビームプリンタ（以下、LBPと略す）の内部構造を示す断面図で、このLBPは、文字パターンデータ等を入力して記録紙に印刷することができる。

【0039】図において、740はLBP本体であり、供給される文字パターン等を基に、記録媒体である記録紙上に像を形成する。700は操作のためのスイッチ及びLED表示器などが配されている操作パネル、701はLBP740全体の制御及び文字パターン情報等を解析するプリンタ制御ユニットである。このプリンタ制御ユニット701は主に文字パターン情報をビデオ信号に変換してレーザドライバ702に出力する。

【0040】レーザドライバ702は半導体レーザ703を駆動するための回路であり、入力されたビデオ信号に応じて半導体レーザ703から発射されるレーザ光704をオン・オフ切替える。レーザ光704は回転多面鏡705で左右方向に振られて静電ドラム706上を走査する。これにより、静電ドラム706上には文字パターンの静電潜像が形成される。この潜像は静電ドラム706周囲の現像ユニット707により現像された後、記録紙に転写される。この記録紙にはカットシートを用

い、カットシート記録紙はLBP740に装着した用紙カセット708に収納され、給紙ローラ709及び搬送ローラ710と711とにより装置内に取込まれて、静電ドラム706に供給される。

【0041】尚、エンジン213はLBPに限定されるものでなく、以下で説明するインクジェットプリンタ等も適用可能である。

〈装置本体の概略説明〉図7は、エンジン213として利用できるインクジェット記録装置IJRAの概略図である。同図において、駆動モータ5013の正逆回転に連動して駆動力伝達ギア5011、5009を介して回転するリードスクリュー5005の螺旋溝5004に対して係合するキャリッジHCはピン（不図示）を有し、矢印a、b方向に往復移動される。このキャリッジHCには、インクジェットカートリッジIJCが搭載されている。5002は紙押え板であり、キャリッジの移動方向に互って紙をプラテン5000に対して押圧する。5007、5008はフォトカブラで、キャリッジのレバー5006のこの域での存在を確認して、モータ5013の回転方向切り換え等を行うためのホームポジション検知手段である。5016は記録ヘッドの前面をキャップするキャップ部材5022を支持する部材で、5015はこのキャップ内を吸引する吸引手段で、キャップ内開口5023を介して記録ヘッドの吸引回復を行う。5017はクリーニングブレードで、5019はこのブレードを前後方向に移動可能にする部材であり、本体支持板5018にこれらが支持されている。ブレードは、この形態でなく周知のクリーニングブレードが本例に適用できることは言うまでもない。又、5021は、吸引回復の吸引を開始するためのレバーで、キャリッジと係合するカム5020の移動に伴って移動し、駆動モータからの駆動力がクラッチ切り換え等の公知の伝達手段で移動制御される。

〈上記印刷システムの変形例〉上記例では、PDLドライバをデフォルトのドライバとして取扱っていたが、ホストコンピュータの性能が非常に高く、メモリ等のホスト側の資源が豊富にあり、通信回線が高速なものであるとプリンタドライバが判定した場合、自動的にデフォルトのドライバをラスタドライバとしてアプリケーションからの描画を行うこともできる。

【0042】この判定は、プリンタドライバのインストール時、あるいはホスト側の環境が変更された時点で行われる。

【0043】この方法を採用すれば、ラスタドライバがPDLドライバと比べて常に同等かそれ以上のパフォーマンスを出すホスト・プリンタ環境においては、PDLデータ生成からラスタデータ生成への処理の移行を行う必要がなくなるため、処理全体を効率化することができる。

〔第2の実施形態〕第2の実施の形態を説明する。本実施の形態は、第1の実施の形態と同じ構成であり、ホス

トコンピュータにおける図3及び図5の手順をそれぞれ図8及び図9に置き換えたものである。

【0044】図8、図9は、プリンタドライバ102により行われる処理の一部であり、PDLドライバとラス  
タドライバといずれを用いるか選択して印刷出力を行な  
わせるものである。図3の処理は、アプリケーションから  
の描画コマンドごとに行われる処理であるが、図8の  
処理は1ページ分についてまとめて行われる点で異なっ  
ている。すなわち、図9においてPDLを選択するかラス  
タを選択するかは、処理時間の予測を1ページ分の描  
画コマンドに対して行い、その結果を比較して決定され  
る。

【0045】まず、ステップS801においてプリンタ  
へ転送するデータの作成をホストコンピュータ上で開始  
する。ステップS802では、アプリケーション101  
より出力されるグラフィックインターフェースの描画コ  
マンドを受け取る。

【0046】ステップS803においては、アプリケー  
ションから出力された描画コマンドをドライバで変換し  
てプリンタへ転送する際に、PDLデータとして転送し  
た方が良いのか、ラスタデータとして転送した方が良い  
のかを判定する。ステップS803の判定アルゴリズム  
の詳細を図9に示す。

【0047】まず、ステップS901において、グラ  
フィックスインターフェースの転送する描画コマンドの中  
に、PDLでは描画不可能なものが含まれているかを判  
定する。

【0048】PDLで描画不可能とは、処理対象の描画  
コマンドを、それをラスタライズした場合と正確に同じ  
結果を産み出すPDLに変換することができない、とい  
うことである。PDLで描画不可能な描画コマンドはそ  
れらの組合わせしだいで予め知ることができる。そのた  
め、プリンタドライバのインストール時、あるいはアプリ  
ケーションのインストール時など、描画コマンドとP  
DLとの組み合わせが変わる場合に、組合せごとにPD  
Lで描画不可能な描画コマンドを登録しておく。そうす  
ることによって、PDLでは描画不可能として登録された描  
画コマンドを参照してステップS901における判定を下  
すことができる。

【0049】PDLで描画不可能と判定されれば、ステ  
ップS807へ進み、ラスタドライバに処理を切替え  
る。

【0050】PDLですべて描画可能と判定された場合  
は、次にステップS902へ進み、処理対象となってい  
る1ページ分のグラフィックスインターフェースコマン  
ドをPDLに置き換えた場合に、プリンタによってどの  
位の描画時間がかかるのかを予測計算する。予測計算に  
は、各PDLコマンドを様々なパラメータで実行した場  
合の計算コストを示す、予め準備された情報を利用す  
る。この情報に、処理対象となっている描画コマンドの

パラメータを適用した場合の計算コストを描画に必要な  
時間として予測する。

【0051】次にステップS903において、処理対象  
となっている1ページ分のグラフィックスインターフェ  
ースコマンドをラスタドライバを利用してホストコンピ  
ュータでラスタライズする時間を予測計算する。予測計  
算には、様々なパラメータで送られる各グラフィックス  
インターフェイスコマンドをラスタライズした場合の計  
算コストを示す、予め準備された情報を利用する。この  
場合にも、ステップS902と同様、処理対象となってい  
る描画コマンドのパラメータを適用して時間を予測す  
る。なお、ラスタ形式のデータを圧縮してプリンタに送  
付し、プリンタで伸長する場合には、その圧縮・伸張に  
要する時間も予測計算する。

【0052】ステップS904において、PDL予測描  
画時間とグラフィックスインターフェースコマンドラス  
タライズ予測描画時間とを比較し、ラスタライズ予測時  
間の方が短い場合、ステップS806へ進み、ラスタド  
ライバへ処理を切替える。

【0053】一方、PDL描画時間の方が短い場合には  
ステップS804へ進み、PDLデータを作成する。

【0054】ステップS805では、アプリケーション  
から転送される描画コマンドが完了したかをチェッ  
クし、完了していると判定されれば生成されたPDLデー  
タをプリンタに転送する。完了していなければ次のペー  
ジの描画コマンドについて処理を施す。

【0055】一方、ステップS806へと進み、ラス  
タドライバへの切替えが発生した場合、プリンタドライバ  
内部でPDLドライバからラスタドライバへの切替えを  
行い、ラスタデータとしてプリンタへデータ転送を行う  
処理に切り替える。

【0056】ステップS807において、処理対象とな  
っているグラフィックスインターフェースコマンドをラ  
スタドライバによって解析し、ラスタデータを生成す  
る。

【0057】この後、ステップS808で再びプリン  
タドライバをラスタドライバからPDLドライバに戻して  
おく。

【0058】こうして変換されたページごとのPDLデ  
ータ、あるいはラスタデータを、ステップS809でプ  
リンタに転送する。この際、ラスタデータは、必要に応  
じて圧縮される。プリンタでは、圧縮されている場合に  
はそのデータを伸張圧縮ハードウェア211により伸長  
する。

【0059】以上の手順により、ホストコンピュータ上  
のアプリケーションで作成された画像を印刷出力する  
際、ラスタデータあるいはPDLデータのどちらの形式  
のデータとして印刷データをプリンタに送信するかを決  
定し、決定された形式に変換してデータをプリンタに送  
信する。このため、データに応じて最適な形式を選ぶこ

とができ、印刷効率を向上させることができる。

【0060】本実施の形態ではページ単位に形式を決定するため、PDL形式の方が効率がよいページとラスタ形式の方が効率がよい、あるいはラスタ形式でなければ再現できないページがもじったデータであっても、迅速に印刷することができる。

【0061】

【他の実施形態】なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0062】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読出し実行することによっても、達成されることは言うまでもない。

【0063】この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0064】プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0065】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0066】さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0067】本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになるが、簡単に説明すると、図10のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。

【0068】すなわち、少なくとも、画像データをラスタ形式に変換するための第1の時間を予測する第1の予

測工程のコードと、ページ記述言語形式に変換された前記画像データを印刷装置により展開するための第2の時間を予測する第2の予測工程のコードと、第1の時間と第2の時間とを比較し、より短時間である方の形式に前記描画命令で記述された画像データを変換する変換工程のコードと、変換工程により変換されたデータを前記印刷装置に送信する送信工程のコードの各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0069】

10 【発明の効果】以上説明したように、本発明に係る印刷制御方法及び印刷システムは、印刷しようとするデータに適した形式で印刷データを印刷装置に送り、高画質かつ効率の良い印刷処理を行なえるという効果を奏する。

【0070】

【図面の簡単な説明】

【図1】本実施形態のホストコンピュータの印刷制御関連プログラムの基本構成を示すブロック図である。

【図2】本実施形態のプリンタの基本構成を示すブロック図である。

20 【図3】ホストコンピュータによる印刷制御手順のフローチャートである。

【図4】ホストコンピュータによる印刷制御手順のフローチャートである。

【図5】本実施形態のホストコンピュータの構成を示すブロック図である。

【図6】レーザビームプリンタの断面図である。

【図7】インクジェットプリンタの斜視図である。

【図8】第2の実施形態のホストコンピュータによる印刷制御手順のフローチャートである。

30 【図9】第2の実施形態によるホストコンピュータによる印刷制御手順のフローチャートである。

【図10】第1の実施の形態のホストコンピュータの制御を実現するプログラムの記憶媒体上でのメモリマップである。

【符号の説明】

101 アプリケーション

102 プリンタドライバ

103 PDLドライバ

104 ラスタドライバ

40 201 ホスト計算機

202 データ入力用バッファ

203 フォントROM

204 パネルI/Oプロセッサ

205 拡張I/F

206 プログラムROM

207 管理用RAM

208 ハードウェアレンダラ

209 ページ（バンド）バッファ

210 プリンタインターフェイス

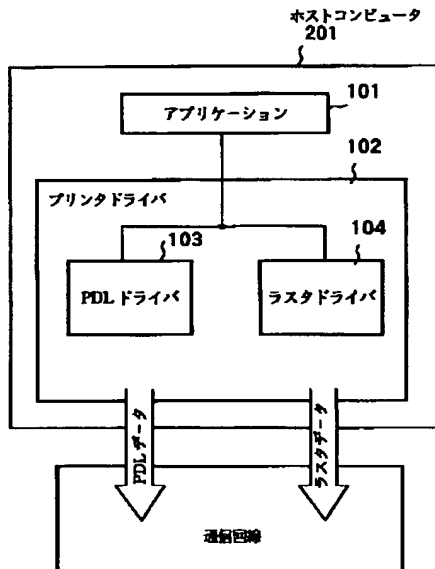
50 211 圧縮・伸長ハードウェア



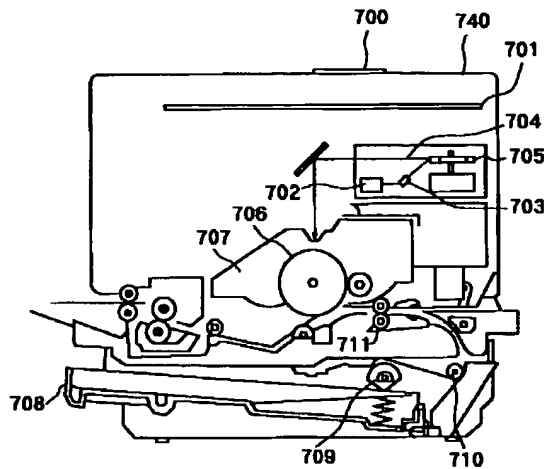
212 CPU  
213 プリントエンジン

214 コントローラ

【図1】



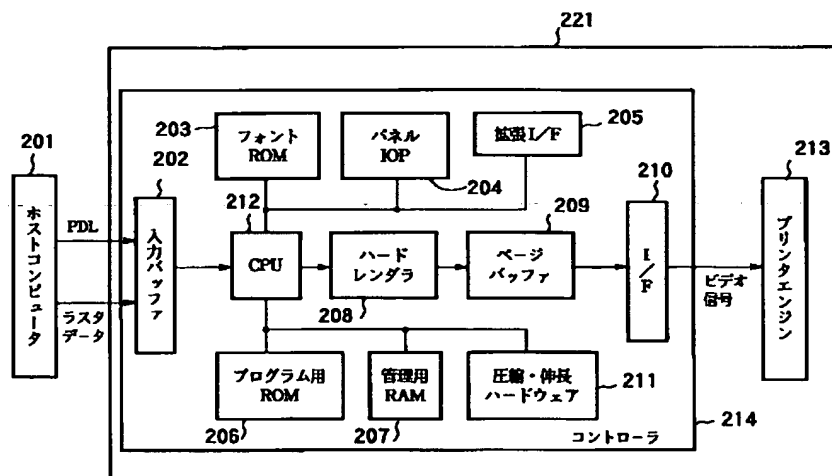
【図6】



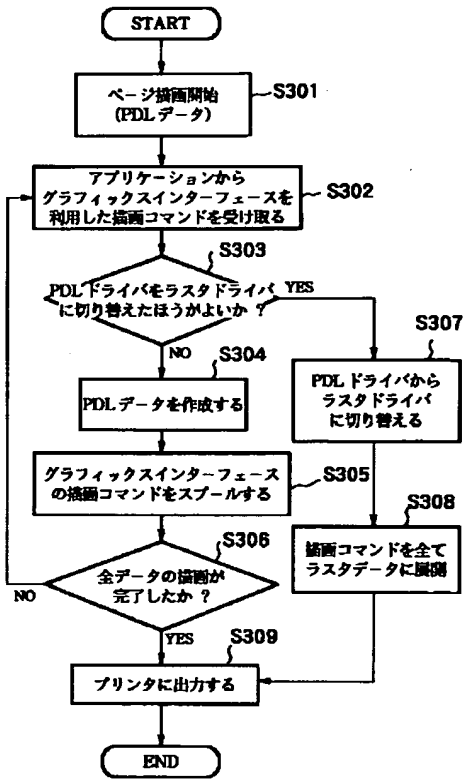
【図10】

ディレクトリ
⋮
第1の予測工程のコード
第2の予測工程のコード
変換工程のコード
送信工程のコード

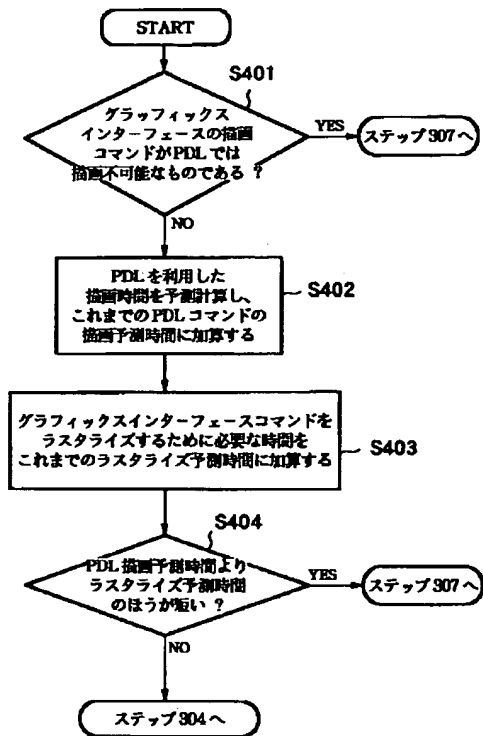
【図2】



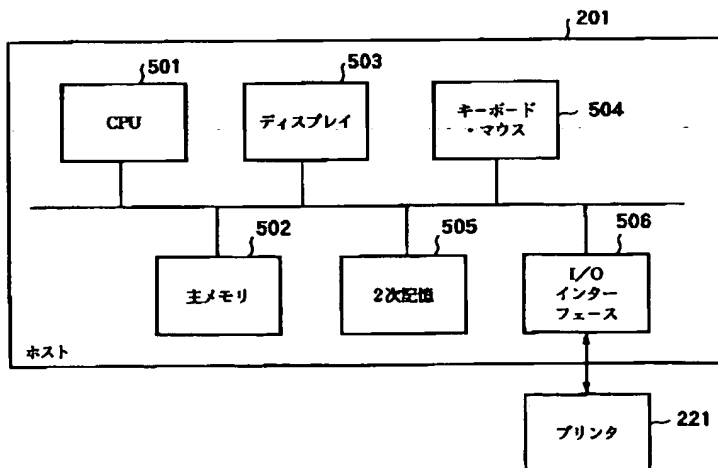
【図3】



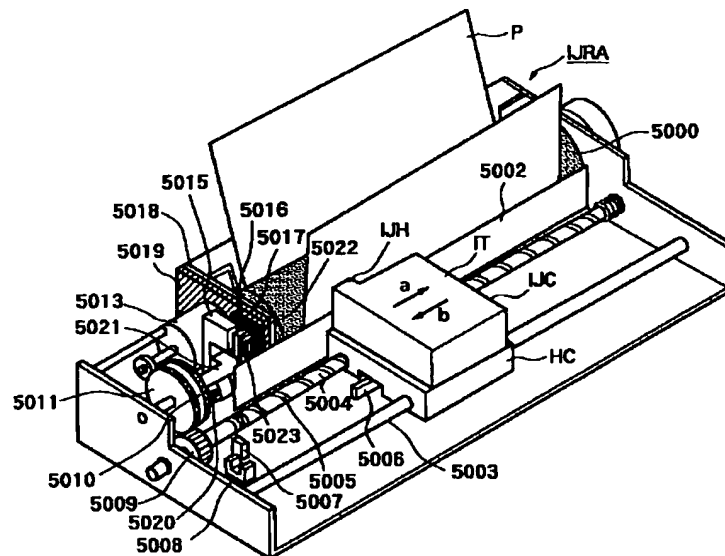
【図4】



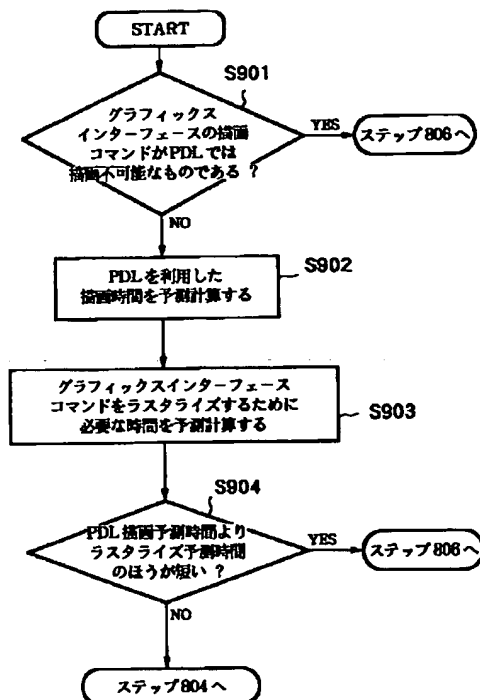
【図5】



【图7】



【図9】



【図8】

